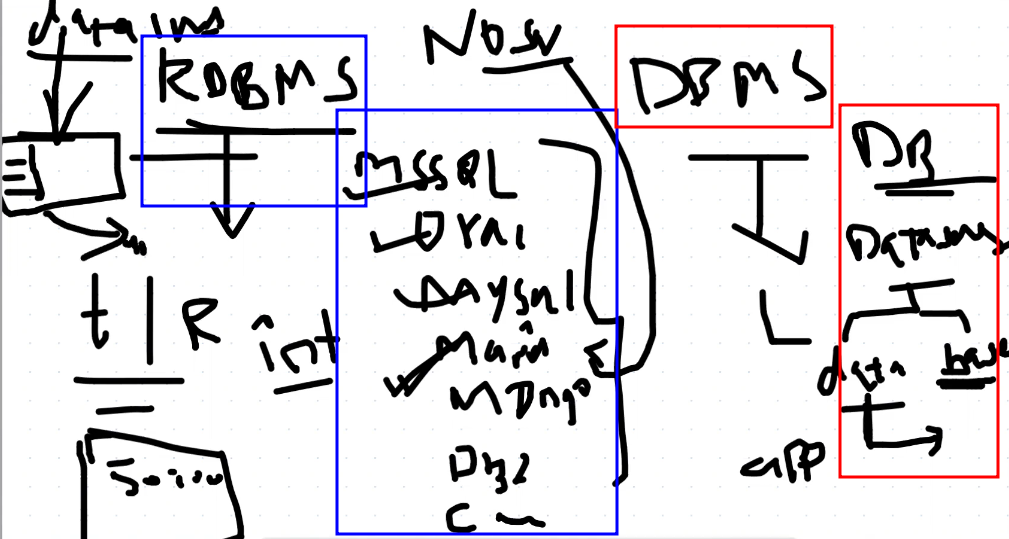
**Lecture 23**

MySQL - RDBMS Part1 (Live Session 24 September 2022)

What is database?

Two words

* Data
* Base



DBMS (Database Management System) and RDBMS (Relational Database Management System) are both software systems that are used to manage databases. However, there are some fundamental differences between the two.

**DBMS** is a software system that enables users to define, create, maintain and control access to a database. It provides various facilities for data storage, retrieval, and management. DBMS may or may not use a ***structured approach*** (Structured approach for data organization and storage refers to a method of organizing data in a way that makes it easier to manage, retrieve, and manipulate. In this approach, data is stored in a predefined structure such as tables, columns, and rows, and it follows a specific set of rules for organization and storage) for data organization and storage.

On the other hand, **RDBMS** is a type of DBMS that uses a ***relational model*** (The relational model is a data model used by Relational Database Management Systems (RDBMS) to organize and store data. It is based on the idea of representing data as a set of related tables, where each table represents an entity or a concept and each row represents an instance of that entity. In the relational model, data is organized into tables that consist of columns and rows. The columns represent the attributes of the entity or concept being modeled, and each row represents a specific instance of the entity with values for each attribute.) for data organization and storage. In an RDBMS, data is stored in tables, with each table having a unique identifier (primary key) and relationships established between tables through foreign keys. An RDBMS enforces referential integrity, which means that the relationships between tables are maintained, and the database remains consistent.

In summary, while both DBMS and RDBMS are software systems that manage databases, RDBMS specifically refers to a type of DBMS that uses a relational model for data organization and storage.

**Explain with examples "attributes of the entity or concept being modeled,"**

Attributes of an entity or concept in the context of the relational model are the characteristics that describe the entity or concept being modeled. These attributes are represented by columns in a table, and each row in the table represents an instance of that entity with specific values for each attribute.

For example, consider a table that represents a customer entity in a retail database. The attributes of the customer entity could include:

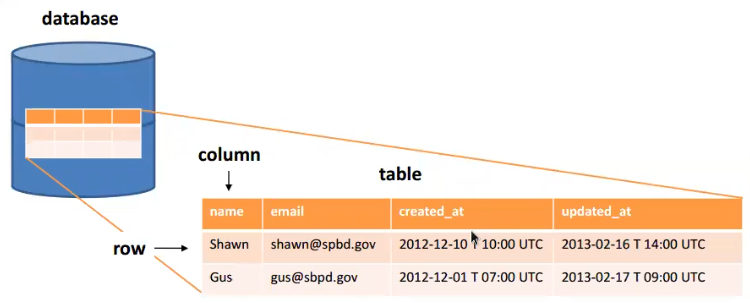
* Customer ID: a unique identifier for each customer
* First Name: the first name of the customer
* Last Name: the last name of the customer
* Email Address: the email address of the customer
* Phone Number: the phone number of the customer
* Address: the address of the customer
* City: the city where the customer resides
* State: the state where the customer resides
* Postal Code: the postal code of the customer's address
* Country: the country where the customer resides

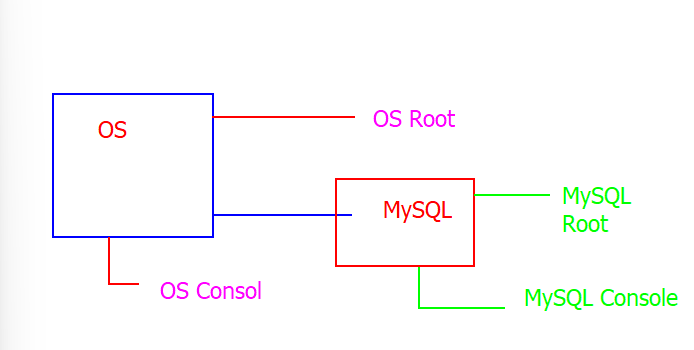
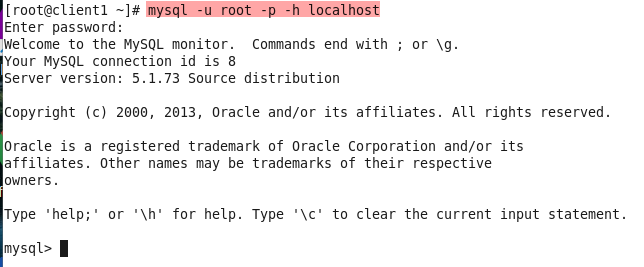
In this example, the customer entity is being modeled with 10 attributes that describe various aspects of the customer's information. Each row in the table represents a specific customer with values for each attribute, such as:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Customer ID** | **First Name** | **Last Name** | **Email Address** | **Phone Number** | **Address** | **City** | **State** | **Postal Code** | **Country** |
| 1001 | John | Smith | [john@example.com](mailto:john@example.com) | 555-123-4567 | 123 Main St | Dallas | TX | 75001 | USA |
| 1002 | Jane | Doe | [jane@example.com](mailto:jane@example.com) | 555-987-6543 | 456 Oak St | Austin | TX | 78701 | USA |

In this table, each row represents a specific customer, and each column represents an attribute of the customer entity, such as First Name, Last Name, and Email Address. By organizing the customer data in this way, it becomes easier to retrieve and manipulate the data in a structured and efficient manner.

In CentOS 7 and further versions 🡪 mySQL 🡪 rename asMariaDB



* Installation
* **Check** 🡪 rpm -qa | grep -I <package>
* $ yum install mariadb-\* -y
* Geekstupp.com 🡪 recommended website for installation
* 
* ***MySQL root is Admin of MySQL 🡪 MySQL port is 3306***
* For the first time connection to MySQL 🡪 MySQL root requires to set its password
* **To set MySQL password** 🡪 a utility called “MySQL Admin” is used to set the root password from OS.
* $ mysqladmin -u root password <New\_password> 🡪 to set root account of
* *$ mysqladmin -u root password 1234*
* MySQL so that database may be connected.
* After password 🡪 mysql console will open
* **mysql>**
* 
* $ mysql -u root -p -h localhost 🡪 to connect to local machine.
* Or
* $ mysql -u root -p -h <SQL\_Server\_IP> 🡪 to connect to SQL server remotely from outside.
* **Tip:-** to open a specific Port 🡪 ***open the service start the port.***
* mysql> help 🡪 to het help about flags etc.
* mysql> show databases; 🡪 to display available databases.
* A picture containing text

  Description automatically generated
* **Information\_schema** 🡪

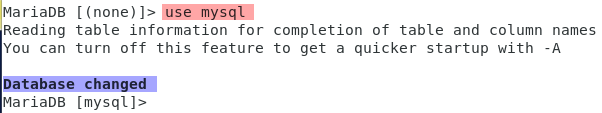
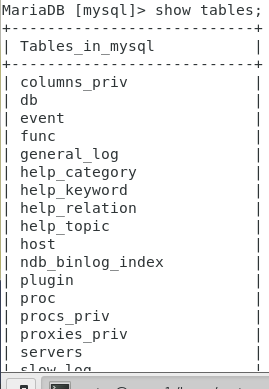
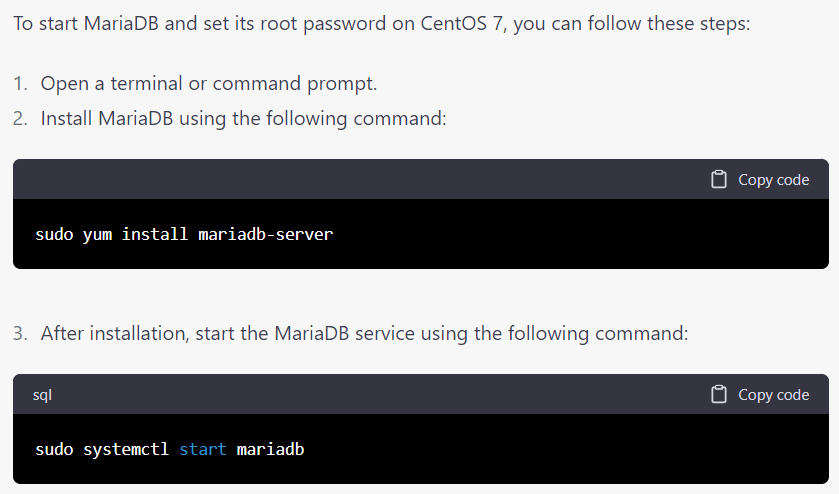
The information schema is a virtual database in MariaDB (and other database management systems) that provides access to metadata, such as information about database objects, including tables, columns, indexes, and constraints. The information schema is designed to be a standard way for users to obtain information about the structure of their databases, and it is defined by the SQL standard.

The information schema is made up of a set of tables that contain information about the database objects and their attributes. These tables can be queried like regular tables using SQL commands, and the results of the queries return information about the database's metadata.

Some of the common uses of the information schema include:

* Retrieving information about database objects such as tables, columns, and indexes.
* Examining the privileges and access rights of database users.
* Monitoring database performance by querying system statistics.
* Obtaining information about triggers, views, and stored procedures.

In summary, the information schema provides a standardized way to access metadata about a MariaDB database, making it easier for developers and administrators to manage the database and its associated objects.

* *Information schema and mysql* 🡪 default databases 🡪 configuration of “mysql”
* mysql> **use database** <database> 🡪 to use a specific database.
* mysql>use mysql 🡪 to use a database.
* 
* mysql> show tables;
* 
* .
* To drop a database 🡪 to delete a database
* > ***drop database <database\_name>.***
* .
* Installation Process of MariaDB in CentOS 7
* 
* Graphical user interface, application, Teams

  Description automatically generated
* Graphical user interface, application

  Description automatically generated
* Graphical user interface, text, application

  Description automatically generated